

### EXERCISE 1:

```
x = 72
y = 3.0
tup = (x, y, x + y, x - y, x / y)
print(tup)
type(tup) # tup is of type tuple
tup[2] = "hello" # tuples are immutable, so this will give us an error
```

### OUTPUT:

```
(72, 3.0, 75.0, 69.0, 24.0)
```

```
-----
TypeError                                Traceback (most recent call last)
<ipython-input-9-c9636d3fc3f2> in <module>()
      4 print(tup)
      5 type(tup) # tup is of type tuple
----> 6 tup[2] = "hello" # tuples are immutable, so this will give us an error
```

```
TypeError: 'tuple' object does not support item assignment
```

### EXERCISE 2:

#2.1

```
p_lst1 = []
for i in range(1, 100):
    is_p = True
    if i == 1:
        is_p = False
    for j in range(2, i//2 + 1):
        if i % j == 0:
            is_p = False
    if (is_p):
        p_lst1.append(i)
```

```
print(p_lst1)
```

#2.2

```
p_lst2 = [i for i in range(2, 100) if all(i % y != 0 for y in range (2, i))]
print(p_lst2)
```

### OUTPUT:

```
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79,
83, 89, 97]
[2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73, 79,
83, 89, 97]
```

### EXERCISE 3:

```
def isprime(n):
    for i in range(2, n//2 + 1):
        if n % i == 0:
            return False
    return True

myprimes = [i for i in range(1, 100) if isprime(i)]
print(myprimes)
```

OUTPUT:

```
[1, 2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73,
79, 83, 89, 97]
```

EXERCISE 4:

```
unit_matrix = np.ones(shape=(5,5))
random_matrix = np.random.rand(5,5)
print(random_matrix * unit_matrix)
```

OUTPUT:

```
[[0.43982567 0.12629943 0.49965636 0.72772329 0.31253183]
 [0.29189846 0.59377083 0.66000806 0.54653244 0.97892983]
 [0.59841291 0.6404166 0.81458493 0.39409193 0.76595933]
 [0.37569559 0.96448683 0.15864885 0.881794 0.48866046]
 [0.63931998 0.21316691 0.23267145 0.00216676 0.60586826]]
```

EXERCISE 5:

```
for key, value in my_dict.items():
    if (value % 2 != 0):
        print(key)
        my_dict[key] = value * 2

print(my_dict)
```

OUTPUT:

```
1
2
3
{1: 2.0, 2: 6.0, 3: 10.0, 4: 4.0, 5: 2.0}
```